

BBC RD 1974/7



RESEARCH DEPARTMENT

REPORT

---

**HADAMARD TRANSFORMATION:  
a real-time transformer  
for broadcast standard  
p.c.m. television**

R. Walker, B.Sc.(Eng.)



HADAMARD TRANSFORMATION: A REAL-TIME TRANSFORMER  
FOR BROADCAST STANDARD P.C.M. TELEVISION  
R. Walker, B.Sc.(Eng.)

**Summary**

*The requirements and design of a modular Walsh-Hadamard transformer for the real-time transformation of broadcast quality p.c.m. television are discussed.*

*Although the example given is for a transform size of 32 picture elements, the modular approach allows any size of transformer to be constructed, using similar units.*

*This experimental transform system includes the means by which the data describing the transform might be modified in order that investigations into the properties and behaviour of the Walsh-Hadamard transform may be carried out, together with facilities for causing random errors in the transform signal.*

*The modular approach was shown to be satisfactory, but some modifications would be necessary in order to allow changes in the transform-signal control-program to be made quickly enough to permit meaningful subjective tests to be carried out.*

Issued under the authority of



Head of Research Department

Research Department, Engineering Division,  
BRITISH BROADCASTING CORPORATION

February 1974

(PH-120)



# HADAMARD TRANSFORMATION: A REAL-TIME TRANSFORMER FOR BROADCAST STANDARD P.C.M. TELEVISION

Section	Title	Page
<b>Summary</b>	.....	<b>Title Page</b>
<b>1.</b> <b>Introduction</b>	.....	<b>1</b>
<b>2.</b> <b>Theory</b>	.....	<b>1</b>
<b>2.1.</b> <b>Theory of the Walsh-Hadamard Transform</b>	.....	<b>1</b>
<b>2.2.</b> <b>The Fast Walsh-Hadamard Transform algorithm</b>	.....	<b>1</b>
<b>3.</b> <b>Design of the transformer stages</b>	.....	<b>2</b>
<b>4.</b> <b>The complete Walsh-Hadamard transform system</b>	.....	<b>3</b>
<b>4.1.</b> <b>Representation of negative numbers</b>	.....	<b>3</b>
<b>4.2.</b> <b>The complete transform assembly</b>	.....	<b>4</b>
<b>4.3.</b> <b>The control system</b>	.....	<b>5</b>
<b>5.</b> <b>Transform – data control system</b>	.....	<b>5</b>
<b>6.</b> <b>Performance</b>	.....	<b>6</b>
<b>7.</b> <b>Conclusions</b>	.....	<b>6</b>
<b>8.</b> <b>References</b>	.....	<b>6</b>

© BBC 2004. All rights reserved. Except as provided below, no part of this document may be reproduced in any material form (including photocopying or storing it in any medium by electronic means) without the prior written permission of BBC Research & Development except in accordance with the provisions of the (UK) Copyright, Designs and Patents Act 1988.

The BBC grants permission to individuals and organisations to make copies of the entire document (including this copyright notice) for their own internal use. No copies of this document may be published, distributed or made available to third parties whether by paper, electronic or other means without the BBC's prior written permission. Where necessary, third parties should be directed to the relevant page on BBC's website at <http://www.bbc.co.uk/rd/pubs/> for a copy of this document.

# HADAMARD TRANSFORMATION: A REAL-TIME TRANSFORMER FOR BROADCAST STANDARD P.C.M. TELEVISION

R. Walker, B.Sc.(Eng.)

## 1. Introduction

The use of orthogonal transformation<sup>1,2,3,4,5</sup> for picture coding and subsequent bit-rate reduction has been the subject of much discussion (Reference 5 gives a particularly comprehensive bibliography). This report describes the design and construction of a digital real-time Walsh-Hadamard 'transformer'<sup>6,7,8</sup> and inverse transformer for broadcast standard p.c.m. colour television. Such a signal involves a sampling rate of approximately 13.5 MHz ( $\approx 2.4 \times$  Nyquist rate) and a word length of 8 bits.

The transformer design uses a modular system which can be easily modified in both word-length and transform block-size so as to suit a wide range of requirements. The example described is suitable for a transform block-size of 32 and an input (and output) word length of 8 bits. For experimental reasons, all the bits generated by the transform process were retained. The transform domain word length was therefore 13 bits plus a sign bit.

## 2. Theory

### 2.1. Theory of the Walsh-Hadamard Transform

An orthogonal matrix whose normalised elements can take only the values +1 or -1 is known as a Hadamard matrix. One such matrix, of order 2<sup>\*</sup> is

$$H_2 = \begin{vmatrix} 1 & 1 \\ 1 & -1 \end{vmatrix} \quad (1)$$

Repeated application of the recurrence equation:

$$H_{2n} = \begin{vmatrix} H_n & H_n \\ H_n & -H_n \end{vmatrix} \quad (2)$$

gives a particular subset of Hadamard matrices. This particular set of square matrices has orders in the sequence 1, 2, 4, 8, ..., 2<sup>k</sup>, where k is the set of positive integers. Each row (and column, as the matrix is symmetric) is a one-dimensional Walsh function<sup>9</sup> and the matrix represents the complete orthogonal set of Walsh functions of length 2<sup>k</sup>. If this type of matrix is used as an operator to transform an n-dimensional data vector, where n = 2<sup>k</sup>, the resulting transformation is known as the Walsh-Hadamard transform (or sometimes the Walsh-Fourier or Walsh) or, more loosely, the Hadamard transform.

The process is illustrated for a transformation of order 8, where {ABCDEFGH} is the vector representing the 8 samples from the input data, and where + represents +1 and - represents -1.

+++ + + +	x A	= A + B + C + D + E + F + G + H
+ - + - - +	B	A - B + C - D + E - F + G - H
+ + - - + + -	C	A + B - C - D + E + F - G - H
+ - - + + - - +	D	A - B - C + D + E - F - G + H
+ + + + - - -	E	A + B + C + D - E - F - G - H
+ - + - + - +	F	A - B + C - D - E + F - G + H
+ + - - - + +	G	A + B - C - D - E - F + G + H
+ - - + - + + -	H	A - B - C + D - E + F + G - H

(3)

### 2.2. The Fast Walsh-Hadamard Transform algorithm

The instrumentation of the transformer is based on a Fast Transform algorithm\* which minimises the storage of intermediate results and also permits the use of exclusively serial-access data storage<sup>10</sup> (shift registers). This algorithm is comparable with the widely used Fast Fourier-Transform algorithm.<sup>11,12,13,14</sup>

Fig. 1 shows how the Fast Walsh-Hadamard Transform (F.W.-H.T.) would be implemented using this algorithm, for a sampled analogue system and a transform order of 8. The processing is carried out in 3 stages, each stage being a combination of a unique storage block and a common arithmetic block.

For a modular system, this division can be implemented at the printed circuit board 'level', so that a complete stage can be made up from a standard arithmetic

\* From an original idea by J.P. Chambers

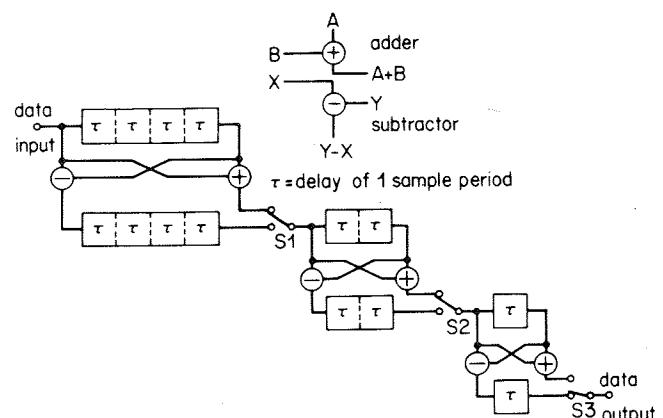


Fig. 1 - Implementation of FW-HT algorithm for 8-dimensional data vector

\* It is interesting to note that the coding of stereo for radio into the two channels R+L and R-L can be represented as a simple Walsh-Hadamard transformation of the two channels R and L.

TABLE 1

Clock Pulse No.	Input Samples	Output of 1st Stage	Output of 2nd Stage	Output of 3rd Stage
1	A			
2	B			
3	C			
4	D			
5	E	A + E		
6	F	B + F		
7	G	C + G	(A+E) + (C+G)	
8	H	D + H	(B+F) + (D+H)	[(A+E) + (C+G)] + [(B+F) + (D+H)]
9		A - E	(A+E) - (C+G)	[(A+E) + (C+G)] - [(B+F) + (D+H)]
10		B - F	(B+F) - (D+H)	[(A+E) - (C+G)] + [(B+F) - (D+H)]
11		C - G	(A-E) + (C-G)	[(A+E) - (C+G)] - [(B+F) - (D+H)]
12		D - H	(B-F) + (D-H)	[(A-E) + (C-G)] + [(B-F) + (D-H)]
13			(A-E) - (C-G)	[(A-E) + (C-G)] - [(B-F) + (D-H)]
14			(B-F) - (D-H)	[(A-E) - (C-G)] + [(B-F) - (D-H)]
15				[(A-E) - (C-G)] - [(B-F) - (D-H)]

board and a special storage board. The design of the storage array can be optimised for the most efficient form of storage, which will be a function of the length of the shift registers required at any particular stage of the transformer.

The process carried out in Fig. 1 is illustrated, together with the contents of the storage units at intermediate steps in the process, by Table 1, assuming that the control inputs of the changeover switches S1, S2, S3 are driven appropriately. The final output can be seen to be identical with the results obtained from Equation 3. Analysis of Table 1 shows that the control signals required are simply the Rademacher functions\*\* from the set of Walsh functions, suitably phased with respect to each other and to the input sample block. These functions can be generated quite simply by a 3-stage binary counter.

To modify this sampled analogue system to a digital form for binary p.c.m., a number of additional points need to be considered. At the sampling-clock frequencies characteristic of video signals in p.c.m. form, the speed requirements dictate that the samples must be processed in the form of parallel binary words which in turn requires that the shift-register stores are several bits 'deep' and the adders, subtractors and data selectors (which replace the analogue changeover switches) have multiple data inputs and outputs. The isometric drawing, Fig. 2, illustrates this arrangement for a single stage of the transformer with shift registers of arbitrary length and a word length of 2 bits. Expansion to  $m$  bits simply involves expansion of the array in the direction M.

\*\* Subset of a set of Walsh Functions; the system of Rademacher functions ( $r_n(\Theta)$ ,  $n = 0, 1, 2, 3, \dots$ ) is defined as:

$$r_n(\Theta) = \text{sign}(\sin(2^{n+1} \cdot \pi \cdot \Theta)) \quad 0 \leq \Theta \leq 1$$

They are basically a set of square waves with decreasing period.

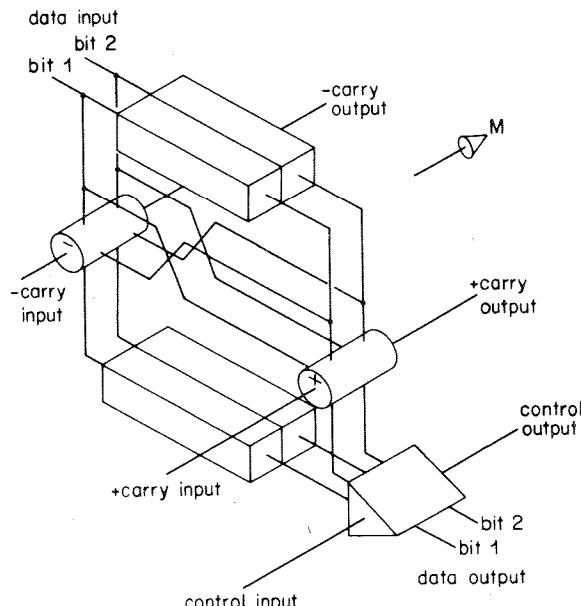
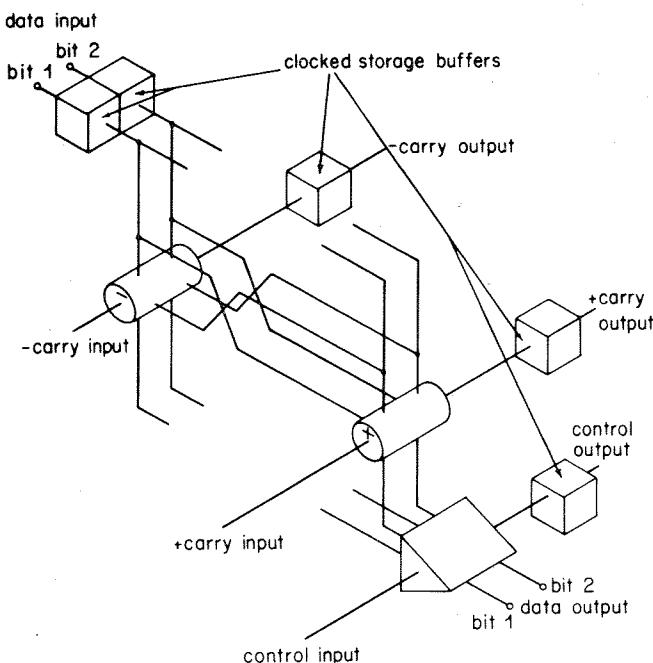


Fig. 2 - Single stage modified for 2-bit digital operation

### 3. Design of the transformer stages

As previously pointed out, each stage of the transformer (and of the inverse transformer, as the matrix is orthogonal) can be divided into two separate parts, the arithmetic logic and the storage array. For a transform of order 32, the required shift-register stages vary in length from 16 bits to 1 bit. As these lengths are readily available in the standard TTL logic 'families', little further comment is required, and the discussion will be restricted to the design of the arithmetic logic. For economic reasons, the TTL families could not be used for shift-register lengths much exceeding 32 bits and a different form of storage would have to be considered, for example, MOS shift registers operating in a multiplexed parallel mode.



*Fig. 3 - Common arithmetic part of each 2-bit stage*

Fig. 3 - shows that part of each of the stages for which the design is common. Operation at a sampling clock (or word) rate of 13.5 MHz allows only 74 ns for each addition (or subtraction) operation. The available time is further reduced by the output propagation delays of the storage elements, the propagation delays of the data selectors and the input set-up time of the following stage. With the adder arrays available at the time of designing the transformer, the remaining part of the clock period was only sufficient to permit the addition of two bits of any word. The additions had, therefore, to be carried out at the rate of two bits per clock pulse, with extra storage elements in the carry output paths to store the carry bit.

This arrangement means that the data is processed throughout the transformer with each pair of bits of any one word being delayed by one clock pulse period relative to the adjacent lower significant pair of bits.

Because of the particular organisation of the logic elements required for the adders and the storage elements, it was convenient to combine two of these two-bit modules together, to give the final module size of 4 bits.

A circuit diagram of this 4-bit module, including an arbitrary length of storage in the signal path is shown in Fig. 4, complete with the necessary clocked input stages and compensating delays in the data-selector control-line. For convenience of p.c. board size, two complete 4-bit modules were arranged on a printed circuit board 15.5 cm wide and 14 cm long. The connections to the main signal path storage units were all brought out at one end of the board to facilitate connection to the storage board. The storage boards were all 15.5 cm and 11.4 cm long, making a total module size of 15.5 cm by 25.4 cm. A photograph of the complete assembly, capable of processing 8-bit

words, is shown in Fig. 5. Any number of these modules may be stacked to give the required word length.

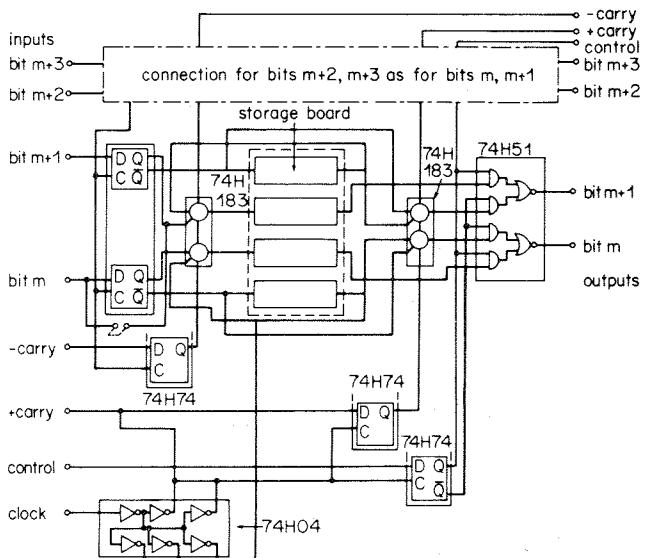
#### 4. The complete Walsh-Hadamard transform system

##### 4.1. Representation of negative numbers

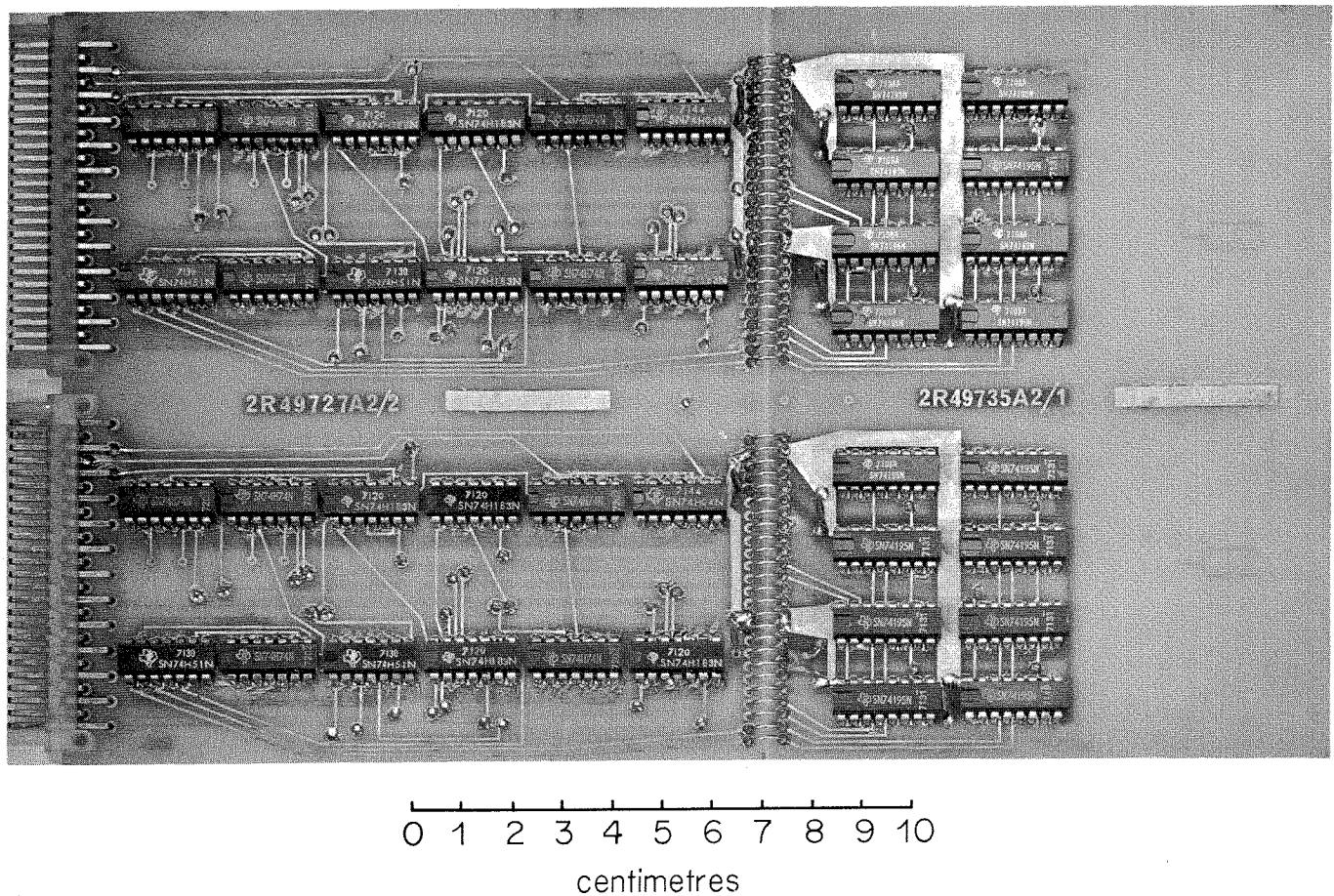
Although the input data are numbers with only positive values (representing the video waveform as numbers increasing from the bottom of the synchronising pulse upwards), negative numbers will inevitably occur during the transform process. Some form of negative number representation is therefore essential. The choice of '2's complement' notation throughout avoids the complexity of adders (and subtractors) designed to use a 'sign and magnitude' notation and also avoid the 'end-around carry' operation necessary with '1's complement' notation.

In 2's complement notation, positive numbers are represented normally, there being a logical zero in the sign-bit location (normally ignored in exclusively positive operations and results); negative numbers have a logical one as the sign bit). In this application, therefore, there was no need to transcode the input data to 2's complement notation, as it is effectively already in the required form.

A problem which arises from the use of 2's complement notation is that the results of an addition or subtraction operation must not be allowed to carry over into the sign bit. Means must be provided for increasing the number of bits at the inputs to an adder or a subtractor at the most significant 'end' of each word; the extra bits inserted must, of course, be numerically zero. In 2's complement notation this can be achieved by making the extra bits the same as the sign bit. Fig. 6 illustrates the principle for an adder with two 4-bit inputs, A1-A4 and B1-B4, and a 5-bit output.



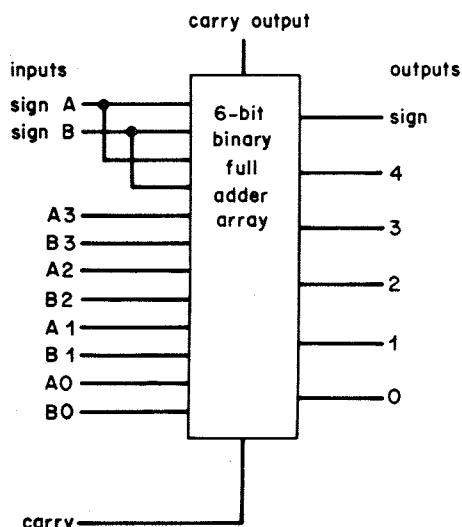
*Fig. 4 - Circuit diagram of 4-bit module*



0 1 2 3 4 5 6 7 8 9 10

centimetres

*Fig. 5 - Complete 8-bit processing board*



*Fig. 6 - Two's complement adder*

In this way, the length of the word can be increased by one bit per stage so that the adder arrays are always just large enough to accommodate the maximum possible word size.

#### 4.2. The complete transform assembly

Fig. 7 illustrates the block diagram of the complete 32 order Walsh-Hadamard transformer, where each of the 4-bit modules described in Section 3 are represented by a block. The input data must be 'pre-skewed' by 1 clock pulse period for every two bits, counting from the least significant. As shown, the transformed data is produced in this skewed form.

As a result of the limit of 8 bits in the final D/A converter, a similar progressive reduction can be applied to the number of bits required at each stage of the inverse transformer as shown in Fig. 8. After the final de-skewing, the 8 bits of the output of the inverse transformer can be converted back to an analogue signal by the D/A converter and thus presented as a picture on a suitable monitor.

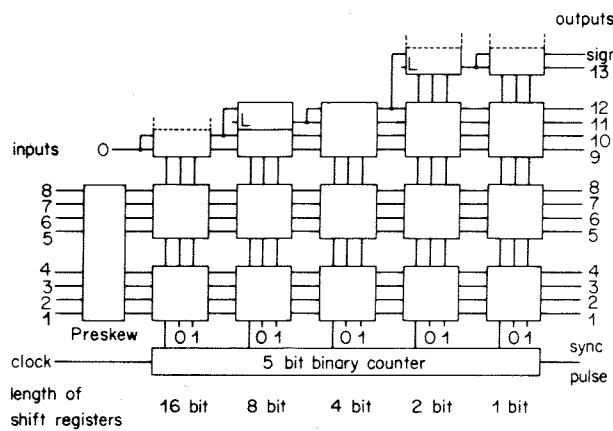


Fig. 7 - Block diagram of  $H_{32}$  transformer

The lengths of shift registers required at each stage of the system are indicated in Figs. 7 and 8. In order to minimise the total storage requirements for the inverse transformer, the order of the stages is reversed as compared to that used in the transformer. This reversal allows the shorter shift registers to be used where the number of bits per word is high and vice versa. A synchronising pulse obtained from the transformer control counter ensures that the inverse-transformer control-counter remains in block synchronism.

Thus, whilst theoretically identical with the transformer, the inverse transformer differs in a number of details.

The final de-skew stage of the inverse transformer additionally contains a protection circuit to guard against the production of data representing negative numbers in the output as a result of truncation or random errors in the transform-domain data. Small negative numbers in the output would otherwise appear as large positive numbers and thus as serious errors. As these negative numbers normally only appear (due to transform data truncation) when the input data is within one or two quantising levels of zero, it is sufficient to output logical zero when they occur.

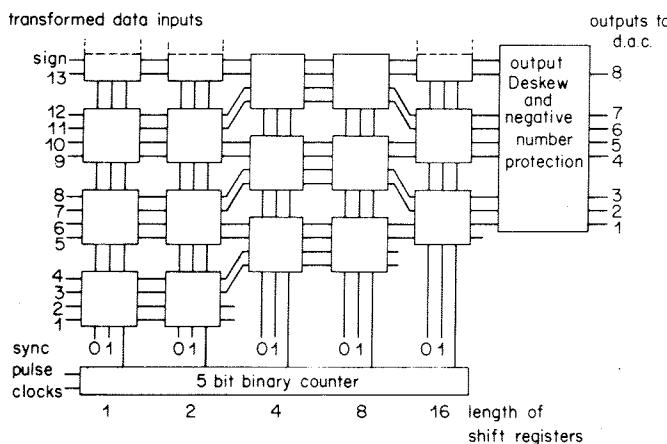


Fig. 8 - Block diagram of inverse  $H_{32}$  transformer

#### 4.3. The control systems

Each of the control counters is simply a 5-stage binary counter with appropriate delays inserted in each of the 5 outputs in order to compensate for the interstage delays in the transformer, and to correct the phases of the counter outputs for the inverse transformer. A synchronising system ensures that these counters remain phase stable relative to each other.

#### 5. Transform – data control system

In order to permit experiments on the data representing the transform, a stage is required, in the transform domain, giving access to and a means of selectively omitting or retaining each bit of each of the words representing the magnitudes of the transform coefficients. As the objective of these experiments was to investigate the effects of omitting part of the transform data, and not to produce a complete bit-rate reduction system, no attempt was made to reduce the maximum data handling capacity of the intermediate transmission path between the transformer and the inverse. This would be done in an operational system, by storing the truncated transform data in a buffer store occurring at an irregular rate as a consequence of the truncation and transmitting this data at a reduced clock rate to a receiving buffer store; here a corresponding process would again provide the irregular truncated transform data for the inverse transformer.

Fig. 9 shows a block diagram of the data control unit. In order to simplify the design of the control stage the output of the transformer is first de-skewed to align all of the bits of each word. The second part of the unit selectively omits or retains the bits of any particular word, according to the instructions in a manually-programmable memory. At each clock pulse the memory steps to the next word in the sequence as the coefficient represented by that word becomes available at the data control logic. Any bit which is to be transmitted is allowed through unaltered and any bit which is to be omitted is changed to the logical state of the sign bit, that is, to a numerical zero.

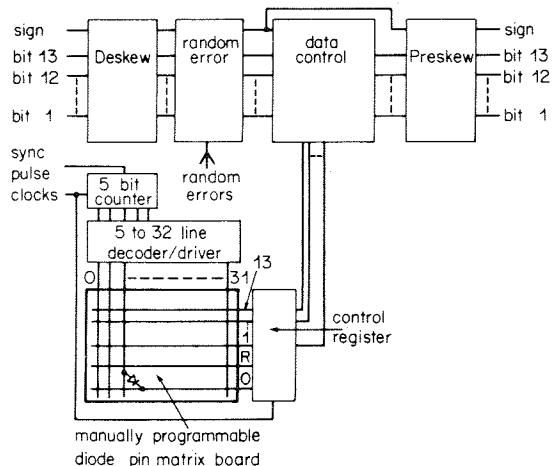


Fig. 9 - The data control unit

The programmable memory used was a diode pin-matrix board with 32 columns representing the transform coefficients in increased order of 'sequency'.\* The 13 rows of the matrix represent the 13 bits of each word. Two additional rows on the matrix board, labelled 'R' and 'O' respectively, permit the retention or omission respectively of all bits of any particular column. This allows entire coefficients to be controlled more simply and is useful for demonstration purposes.

The matrix board is arranged in increasing sequency order for convenience. As the outputs from the transformer do not occur in that order, the drive to the columns of the matrix board have to be temporally re-arranged into the appropriate sequence.

A means of injecting random errors from an external error generator<sup>15</sup> is also included in this part of the system, to permit the investigation of the effects of transmission errors.

Finally, the data is pre-skewed again for compatibility with the inverse transformer input. Fig. 10 shows the important details of the signal path logic.

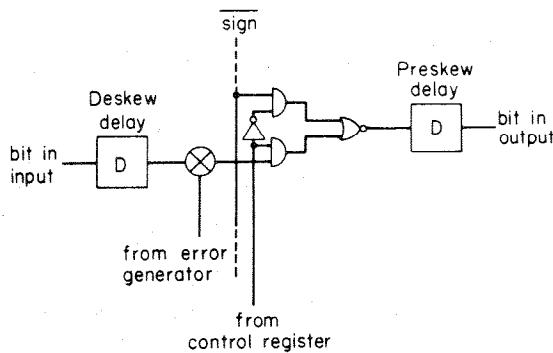


Fig. 10 - Data control unit: signal path logic

## 6. Performance

The output picture as displayed in a picture monitor (with all of the 13 bits and the sign bit retained in the transform domain) showed no degradation as a result of the transformation and subsequent inverse transformation. This is to be expected, since the process was entirely digital, and sufficient capacity was provided to accommodate the results of all the calculations, that is, there were no rounding errors and the output was identical to the input.

## 7. Conclusions

The modular approach to the design of an experimental Walsh-Hadamard transform system has been shown to be

\* 'Sequency' can be defined as the number of zero-crossings in the interval on which the Walsh Functions are defined. It is roughly comparable with frequency in the Fourier domain. An alternative definition, used by Harmuth and others, is half the number of zero-crossings in the interval.

satisfactory. However, the diode pin-matrix board, whilst satisfactory for laboratory investigations into the properties of the transform, is too slow and laborious to change so as to allow the variations necessary if a series of subjective tests is to be carried out readily. A different form of control memory is, therefore, highly desirable for further work to appraise the possibilities for bit-rate reduction.

## 8. References

- HARMUTH, H.F. 1970. *Transmission of information by orthogonal functions*. New York, Springer-Verlag, 1970.
- PRATT, W.K., KANE, J. and ANDREWS, H.C. 1968. Hadamard transform image coding. *Proc. IEEE*, 1969, **57**, 1, pp. 58 – 68.
- LANDAU, H.J. and SLEPIAN, D. 1971. Some computer experiments in picture processing for bandwidth reduction. *Bell Syst. Tech. J.*, 1971, **50**, 5, pp. 1525 – 1540.
- AHMED, N., RAO, K.R. and SCHULTZ, R.B. 1971. The generalized transform. Proc. 1971 Applications of Walsh Functions Symposium, Washington D.C., 1971, pp. 60 – 67.
- WINTZ, P.A. 1972. Transform picture coding. *Proc. IEEE*, 1972, **60**, 7, pp. 809 – 820.
- WELCH, L.R. 1970. Walsh functions and Hadamard matrices. Proc. 1970 Application of Walsh Functions Symposium, Washington D.C., 1970, pp. 163 – 165.
- KENNEDY, J.D. 1971. Walsh function imagery analysis. Proc. 1971 Applications of Walsh Functions Symposium, Washington D.C., 1971, pp. 7 – 10.
- ENOMOTO, H. and SHIBATA, K. 1970. Orthogonal transform coding system for television signals. *J. Inst. TV Eng. of Japan*, 1970, **24**, pp. 99 – 108.
- WALSH, J.L. 1923. A closed set of normal orthogonal functions. *Amer. Journal of Math.*, 1923, **45**, pp. 5 – 24.
- British Complete patent application No. 32837/71.
- COOLEY, J.W. and TUKEY, J.W. 1956. An algorithm for the machine calculation of complex Fourier series. *Math. of Comp.*, 1956, **19**, pp. 297 – 301.
- COOLEY, J.W., LEWIS, P.A.W. and WELCH, P.D. 1967. Historical notes on the fast Fourier transform. *Proc. IEEE*, 1967, **55**, pp. 1675 – 1677.
- BRIGHAM, E.O. and MORROW, R.E. 1967. The fast Fourier transform. *IEEE Spectrum*, 1967, **4**, pp. 6370 – 70.

14. ANDREWS, H.C. 1967. A high speed algorithm for the computer generation of Fourier transforms. *IEEE Trans. Computers*, 1968, C-17, pp. 373 – 375.
15. A digital truly random event generator with a known average output frequency. BBC Research Department Report in course of preparation.